



Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004
Title: Detecting Incorrect Versions of Files

Page 1

S/N 10/821,146

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Cary L. Bates	Examiner: Eric B Kiss
Serial No.: 10/821,146	Group Art Unit: 2192
Filed: April 8, 2004	Confirmation Number: 8972
Title: Detecting Incorrect Versions of Files	Docket: ROC920030419US1

APPEAL BRIEF

**TO THE BOARD OF PATENT APPEALS AND INTERFERENCES
OF THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

This brief is presented in support of the Notice of Appeal filed on March 3, 2008, from the Final Rejection of claims 1 and 3-20 of the above-identified application, as set forth in the Final Office Action mailed on December 3, 2007.

Please charge \$510.00 to Deposit Account 09-0465 to cover the fee for filing an appeal brief. Please charge any additional fees or credit overpayment to Deposit Account 09-0465. Appellant respectfully requests reversal of the Examiner's rejection of pending claims 1 and 3-20.

06/09/2008 LTRUONG 00000043 090465 10821146
01 FC:1402 510.00 DA



Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004
Title: Detecting Incorrect Versions of Files

Page 2

Docket: ROC920030419US1

Table of Contents

1. REAL PARTY IN INTEREST	3
2. RELATED APPEALS AND INTERFERENCES	4
3. STATUS OF CLAIMS	5
4. STATUS OF AMENDMENTS	6
5. SUMMARY OF CLAIMED SUBJECT MATTER	7
6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL.....	21
7. ARGUMENT	22
8. CLAIMS APPENDIX	31
9. EVIDENCE APPENDIX	35
10. RELATED PROCEEDINGS APPENDIX	47



Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004
Title: Detecting Incorrect Versions of Files

Page 3

Docket: ROC920030419US1

1. Real Party in Interest

The real party in interest, in addition to the inventors, Cary L. Bates and Paul W. Buenger, is the assignee, International Business Machines Corporation, a corporation organized and existing under and by virtue of the laws of the State of New York, and having an office and place of business at New Orchard Road, Armonk, New York 10504.

Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004 Docket: ROC920030419US1
Title: Detecting Incorrect Versions of Files

Page 4

2. Related Appeals and Interferences

There are no other prior or pending appeals, interferences, or judicial proceedings, which may be related to, directly affect or be directly affected by, or have a bearing on the Board's decision.

Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004 Docket: ROC920030419US1
Title: Detecting Incorrect Versions of Files

Page 5

3. Status of Claims

On March 3, 2008, appellant appealed from the final rejection of claims 1 and 3-20 made in the Final Office Action dated December 3, 2007. Claim 2 was canceled without prejudice or disclaimer. Finally rejected claims 1 and 3-20 on appeal are set forth in the Claims Appendix.

Appellant: Cary L. Bates

Page 6

Serial No.: 10/821,146

Filed: April 8, 2004

Docket: ROC920030419US1

Title: Detecting Incorrect Versions of Files

4. Status of Amendments

Subsequent to the Final Office Action dated December 3, 2007, appellant did not file any amendment.

5. Summary of Claimed Subject Matter

An embodiment of the invention is described, by way of example and not of limitation, in appellant's specification at page 3, lines 26-28, at page 4, lines 1-3, at Fig. 1, elements 100, 101, 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 3, elements 305 and 320, at Fig. 4, element 425, and at Fig. 5B, elements 560, 565, 570, and 570, which in pertinent part recite:

"A method, apparatus, system, and signal-bearing medium are provided that in an embodiment issue a warning if a file to be used is an older version. In an embodiment, the warning includes an identification of the location of a newer version of the file. In an embodiment, the file is a class, and the old and new versions are found using a classpath, but in other embodiments any type of file or other object may be used."

With reference to claim 1, an embodiment of the invention comprises a method, which is described, by way of example and not of limitation, in the specification, at page 3, line 26, at page 5, lines 17-27, at page 6, lines 1-21, at page 12, lines 21-28, at page 13, lines 1-28, at page 14, lines 1-16, at Fig. 1, elements 100, 101, 102, 168, 170, and 172, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With further reference to claim 1, in an embodiment of the invention, the method comprises finding a first file in a first directory specified in a classpath, which is described, by way of example and not of limitation, in the specification, at page 4, lines 1-3 and 24-25, at page 5, lines 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, element 555.

With further reference to claim 1, in an embodiment of the invention, the method comprises determining whether the first file is an incorrect version, wherein the determining whether the first file is the incorrect version further comprises determining whether a second file later in the classpath from the first file is an earlier version than the first file, which is

described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 1, in an embodiment of the invention, the method comprises if the first file is the incorrect version, issuing a warning, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 3, in an embodiment of the invention, the issuing further comprises: providing an identification of a location of a newer version of the first file, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 22-24, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 11, lines 25-29, at page 12, lines 3-20, at page 13, lines 20-27, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 4, elements 400 and 425, and at Fig. 5B, elements 565 and 570.

With reference to claim 4, in an embodiment of the invention, the determining further comprises: determining whether a second file is owned by a user and the first file is not owned by the user, wherein the second file is later in the classpath than the first file, which is described, by way of example and not of limitation, in the specification, at page 2, lines 27-29, at page 3, lines 1-17 and 26-28, at page 4, lines 1 and 22-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 11, lines 1-29, at page 12, lines 3-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, and 575.

With reference to claim 5, an embodiment of the invention comprises an apparatus comprising various means, which is described, by way of example and not of limitation, in

appellant's specification at page 3, line 26, at page 5, lines 3-27, at page 6, lines 1-28, at page 7, lines 1-28, at page 8, lines 1-4 and 26-28, at page 9, lines 1-28, page 10, lines 1-9, at Fig 1, elements 100, 101, 101A, 101B, 101C, 101D, 102, 103, 104, 105, 111, 112, 113, 114, 121, 122, 123, 124, 125, 126, 127, 128, and 129, which in pertinent part recites:

"A method, apparatus, system, and signal-bearing medium are provided

...

Referring to the Drawing, wherein like numbers denote like parts throughout the several views, Fig. 1 depicts a high-level block diagram representation of a computer system 100, according to an embodiment of the present invention. The major components of the computer system 100 include one or more processors 101, a main memory 102, a terminal interface 111, a storage interface 112, an I/O (Input/Output) device interface 113, and communications/network interfaces 114, all of which are coupled for inter-component communication via a memory bus 103, an I/O bus 104, and an I/O bus interface unit 105.

The computer system 100 contains one or more general-purpose programmable central processing units (CPUs) 101A, 101B, 101C, and 101D, herein generically referred to as the processor 101. In an embodiment, the computer system 100 contains multiple processors typical of a relatively large system; however, in another embodiment the computer system 100 may alternatively be a single CPU system. Each processor 101 executes instructions stored in the main memory 102 and may include one or more levels of on-board cache.

The main memory 102 is a random-access semiconductor memory for storing data and programs. The main memory 102 is conceptually a single monolithic entity, but in other embodiments the main memory 102 is a more complex arrangement, such as a hierarchy of caches and other memory devices. For example, memory may exist in multiple levels of caches, and these caches may be further divided by function, so that one cache holds instructions while another holds non-instruction data, which is used by the processor or processors. Memory may further be distributed and associated with different CPUs or sets of CPUs, as is known in any of various so-called non-uniform memory access (NUMA) computer architectures.

The memory 102 includes a debug controller 168 and a program 172. Although the debug controller 168 and the program 172 are illustrated as being contained within the memory 102 in the computer system 100, in other embodiments some or all of them may be on different computer systems and may be accessed remotely, e.g., via the network 130. The computer system 100 may use virtual addressing mechanisms that allow the programs of the computer system 100 to behave as if they only have access to a large, single storage entity instead of access to multiple, smaller storage entities. Thus, while the debug controller 168 and the program 172 are illustrated as residing in the memory 102, these elements are not necessarily all completely contained in the same storage device at the same time.

The debug controller 168 is used to debug the program 172. The debug controller 168 includes a classpath controller 170. In another embodiment, the classpath controller 170 is separate from the debug controller 168. In another embodiment, the classpath controller 170 is implemented as a class loader or as a portion of a class loader that loads classes that may be used by the program 172. In an embodiment, the classpath controller 170 includes instructions capable of executing on the processor 101 or statements capable of being interpreted by instructions executing on the processor 101 to access or communicate with the user interfaces as further described below with reference to Figs. 2, 3, and 4, and to perform the functions as further described below with reference to Figs. 5A and 5B. In another embodiment, the classpath controller 170 may be implemented in microcode. In yet another embodiment, the classpath controller 170 may be implemented in hardware via logic gates and/or other appropriate hardware techniques, in lieu of or in addition to a processor-based system.

In an embodiment, the program 172 includes instructions or statements capable of being interpreted or compiled to execute on the processor 101. The program 172 may be debugged by the debug controller 168.

The memory bus 103 provides a data communication path for transferring data among the processors 101, the main memory 102, and the I/O bus interface unit 105. The I/O bus interface unit 105 is further coupled to the system I/O bus 104 for transferring data to and from the various I/O units. The I/O bus interface unit 105 communicates with multiple I/O interface units 111, 112, 113, and 114, which are also known as I/O processors (IOPs) or I/O

adapters (IOAs), through the system I/O bus 104. The system I/O bus 104 may be, e.g., an industry standard PCI (Peripheral Component Interconnect) bus, or any other appropriate bus technology. The I/O interface units support communication with a variety of storage and I/O devices. For example, the terminal interface unit 111 supports the attachment of one or more user terminals 121, 122, 123, and 124.

The storage interface unit 112 supports the attachment of one or more direct access storage devices (DASD) 125, 126, and 127 (which are typically rotating magnetic disk drive storage devices, although they could alternatively be other devices, including arrays of disk drives configured to appear as a single large storage device to a host). Various portions of the contents of the DASD 125, 126, and 127 may be loaded and stored from/to the memory 102 as needed.

The I/O and other device interface 113 provides an interface to any of various other input/output devices or devices of other types. Two such devices, the printer 128 and the fax machine 129, are shown in the exemplary embodiment of Fig. 1, but in other embodiment many other such devices may exist, which may be of differing types. The network interface 114 provides one or more communications paths from the computer system 100 to other digital devices and computer systems; such paths may include, e.g., one or more networks 130.

Although the memory bus 103 is shown in Fig. 1 as a relatively simple, single bus structure providing a direct communication path among the processors 101, the main memory 102, and the I/O bus interface 105, in fact the memory bus 103 may comprise multiple different buses or communication paths, which may be arranged in any of various forms, such as point-to-point links in hierarchical, star or web configurations, multiple hierarchical buses, parallel and redundant paths, etc. Furthermore, while the I/O bus interface 105 and the I/O bus 104 are shown as single respective units, the computer system 100 may in fact contain multiple I/O bus interface units 105 and/or multiple I/O buses 104. While multiple I/O interface units are shown, which separate the system I/O bus 104 from various communications paths running to the various I/O devices, in other embodiments some or all of the I/O devices are connected directly to one or more system I/O buses.

...

The computer system 100 depicted in Fig. 1 has multiple attached terminals 121, 122, 123, and 124, such as might be typical of a multi-user "mainframe" computer system. Typically, in such a case the actual number of attached devices is greater than those shown in Fig. 1, although the present invention is not limited to systems of any particular size. The computer system 100 may alternatively be a single-user system, typically containing only a single user display and keyboard input, or might be a server or similar device which has little or no direct user interface, but receives requests from other computer systems (clients). In other embodiments, the computer system 100 may be implemented as a personal computer, portable computer, laptop or notebook computer, PDA (Personal Digital Assistant), tablet computer, pocket computer, telephone, pager, automobile, teleconferencing system, appliance, or any other appropriate type of electronic device.

It should be understood that Fig. 1 is intended to depict the representative major components of the computer system 100 at a high level, that individual components may have greater complexity than that represented in Fig. 1, that components other than or in addition to those shown in Fig. 1 may be present, and that the number, type, and configuration of such components may vary. Several particular examples of such additional complexity or additional variations are disclosed herein; it being understood that these are by way of example only and are not necessarily the only such variations.

The various software components illustrated in Fig. 1 and implementing various embodiments of the invention may be implemented in a number of manners, including using various computer software applications, routines, components, programs, objects, modules, data structures, etc., referred to hereinafter as "computer programs," or simply "programs." The computer programs typically comprise one or more instructions that are resident at various times in various memory and storage devices in the computer system 100, and that, when read and executed by one or more processors 101 in the computer system 100, cause the computer system 100 to perform the steps necessary to execute steps or elements embodying the various aspects of an embodiment of the invention.

Moreover, while embodiments of the invention have and hereinafter will be described in the context of fully functioning computer systems, the various embodiments of the invention are capable of being distributed as a program product in a variety of forms, and the

invention applies equally regardless of the particular type of signal-bearing medium used to actually carry out the distribution. The programs defining the functions of this embodiment may be delivered to the computer system 100 via a variety of signal-bearing media, which include, but are not limited to:

(1) information permanently stored on a non-rewriteable storage medium, e.g., a read-only memory device attached to or within a computer system, such as a CD-ROM readable by a CD-ROM drive;

(2) alterable information stored on a rewriteable storage medium, e.g., a hard disk drive (e.g., DASD 125, 126, or 127) or diskette.”

With further reference to claim 5, in an embodiment of the invention, the apparatus comprises: means for finding a first class in a first directory specified in a classpath, which is described, by way of example and not of limitation, in the specification, at page 4, lines 1-3 and 24-25, at page 5, lines 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, element 555.

With further reference to claim 5, in an embodiment of the invention, the apparatus comprises means for finding a second class in a second directory, wherein the second directory is later in the classpath than the first directory, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 5, in an embodiment of the invention, the apparatus comprises means for determining whether the second class is a newer version of the first class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28,

at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With reference to claim 6, in an embodiment of the invention, the apparatus further comprises means for issuing a warning if the second class is the newer version of the first class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 7, in an embodiment of the invention, the apparatus further comprises: mean for deciding whether the second class is owned by a user and the first class is not owned by the user, which is described, by way of example and not of limitation, in the specification, at page 2, lines 27-29, at page 3, lines 1-17 and 26-28, at page 4, lines 1 and 22-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 11, lines 1-29, at page 12, lines 3-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, and 575.

With reference to claim 8, in an embodiment of the invention, the apparatus further comprises: means for issuing a warning if the second class is owned by the user and the first class is not owned by the user, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 9, an embodiment of the invention comprises: a storage medium encoded with instructions, wherein the instructions when executed, which is described, by way of example and not of limitation, in appellant's specification at page 5,

lines 5-6 and 10-27, at page 6, lines 9-26, at page 7, lines 8-13 and 21-22, at page 9, lines 17-28, at page 10, lines 1-9, at Fig. 1, elements 101, 101A, 101B, 101C, 101D, 102, 125, 126, 127, 168, 170, and 172.

With further reference to claim 9, in an embodiment of the invention, the instructions when executed comprise: finding a first class in a first directory specified in a classpath, which is described, by way of example and not of limitation, in the specification, at page 4, lines 1-3 and 24-25, at page 5, lines 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, element 555.

With further reference to claim 9, in an embodiment of the invention, the instructions when executed comprise finding a second class in a second directory, wherein the second directory is later in the classpath than the first directory, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 9, in an embodiment of the invention, the instructions when executed comprise determining whether the second class is a newer version of the first class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 9, in an embodiment of the invention, the instructions when executed comprise issuing a warning if the second class is the newer version of the first class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6,

lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 10, an embodiment of the invention further comprises: deciding whether the second class is owned by a user and the first class is not owned by the user, which is described, by way of example and not of limitation, in the specification, at page 2, lines 27-29, at page 3, lines 1-17 and 26-28, at page 4, lines 1 and 22-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 11, lines 1-29, at page 12, lines 3-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, and 575.

With reference to claim 11, an embodiment of the invention further comprises: issuing the warning if the second class is owned by the user and the first class is not owned by the user, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 12, an embodiment of the invention further comprises: saving a reason for the warning, which is described, by way of example and not of limitation, in the specification at page 13, lines 25-27 and at Fig. 5B, element 570.

With reference to claim 13, an embodiment of the invention comprises a computer system, which is described, by way of example and not of limitation, in the specification at page 3, line 26, at page 5, lines 3-27, at page 6, lines 1-28, at page 7, lines 1-28, at page 8, lines 1-4 and 26-28, at page 9, lines 1-28, at Fig 1, elements 100, 101, 101A, 101B, 101C, 101D, 102, 103, 104, 105, 111, 112, 113, 114, 121, 122, 123, 124, 125, 126, 127, 128, and 129.

With further reference to claim 13, in an embodiment of the invention, the computer system comprises a processor, which is described, by way of example and not of limitation, in appellant's specification at page 5, lines 5-6 and 10-25, at page 6, lines 22-23 and 25-26, at page 7, lines 21-22, at page 9, lines 17-25, at Fig. 1, elements 101, 101A, 101B, 101C, and 101D.

With further reference to claim 13, in an embodiment of the invention, the computer system comprises memory encoded with instructions, wherein the instructions when executed on the processor, which is described, by way of example and not of limitation, in appellant's specification at page 5, lines 5-6 and 10-27, at page 6, lines 9-26, at page 7, lines 21-22, at page 9, lines 17-28, at page 10, lines 1-9, at Fig. 1, elements 101, 101A, 101B, 101C, 101D, 102, 168, 170, and 172.

With further reference to claim 13, in an embodiment of the invention, the instructions when executed on the processor comprise: finding a first class in a first directory specified in a classpath, which is described, by way of example and not of limitation, in the specification, at page 4, lines 1-3 and 24-25, at page 5, lines 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, element 555.

With further reference to claim 13, in an embodiment of the invention, the instructions when executed on the processor comprise: finding a second class in a second directory, wherein the second directory is later in the classpath than the first directory, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 13, in an embodiment of the invention, the instructions when executed on the processor comprise: deciding whether the second class is owned by a user and the first class is not owned by the user, which is described, by way of

example and not of limitation, in the specification, at page 2, lines 27-29, at page 3, lines 1-17 and 26-28, at page 4, lines 1 and 22-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 11, lines 1-29, at page 12, lines 3-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, and 575.

With reference to claim 14, in an embodiment of the invention, the instructions further comprise: issuing a warning if the second class is owned by the user and the first class is not owned by the user, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 15, in an embodiment of the invention, the issuing further comprises: providing an identification of the second directory, which is described, by way of example and not of limitation, in the specification, at page 12, lines 10-20 and at Fig. 4, element 425.

With reference to claim 16, in an embodiment of the invention, the instructions further comprise: determining whether the second class is a newer version of the first class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 16, an embodiment of the invention further comprises: issuing a warning if the second class is the newer version of the first class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at

page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 17, an embodiment of the invention comprises a method of configuring a computer, which is described, by way of example and not of limitation, in appellant's specification at page 3, line 26, at page 5, lines 3-27, at page 6, lines 1-28, at page 7, lines 1-28, at page 8, lines 1-4 and 26-28, at page 9, lines 1-28, at page 10, lines 1-9 and 13-23, at page 12, lines 21-28, at page 13, lines 1-28, at page 14, lines 1-16, at Fig. 1, elements 100, 101, 101A, 101B, 101C, 101D, 102, 103, 104, 105, 111, 112, 113, 114, 121, 122, 123, 124, 125, 126, 127, 128, and 129, at Fig. 5A, elements 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With further reference to claim 17, in an embodiment of the invention, the method comprises: configuring the computer to find a file in a first directory specified in a classpath, which is described, by way of example and not of limitation, in the specification, at page 4, lines 1-3 and 24-25, at page 5, lines 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, element 555.

With further reference to claim 17, in an embodiment of the invention, the method comprises: configuring the computer to determine whether the file is an older version, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-20, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With further reference to claim 17, in an embodiment of the invention, the method comprises: configuring the computer to issue a warning if the file is the older version, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at

page 10, lines 24-25, at page 11, lines 1-29, at page 12, lines 1-28, at page 13, lines 1-28, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 3, element 320, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, 565, 570, and 575.

With reference to claim 18, in an embodiment of the invention, the method further comprises: configuring the computer to search for a newer version of the file later in the classpath from the older version, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 1-3 and 21-25, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 10, lines 24-25, at page 11, lines 1-18 and 25-26, at page 12, lines 21-28, at page 13, lines 1-24, at page 14, lines 1-6, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 2, element 215, at Fig. 5A, elements 505, 510, 515, 520, 525, and 530, and at Fig. 5B, elements 555, 560, and 575.

With reference to claim 19, in an embodiment of the invention, the warning further comprises: an identification of a location of a newer version of the file, which is described, by way of example and not of limitation, in the specification, at page 3, lines 26-28, at page 4, lines 22-24, at page 5, lines 1-2 and 17-27, at page 6, lines 1-24, at page 11, lines 25-29, at page 12, lines 3-20, at page 13, lines 20-27, at Fig. 1, elements 102, 168, 170, and 172, at Fig. 4, elements 400 and 425, and at Fig. 5B, elements 565 and 570.

With reference to claim 20, in an embodiment of the invention, the file comprises a class, which is described, by way of example and not of limitation, in the specification, at page 3, lines 2-8, at page 4, lines 1 and 24, at page 6, lines 11-13, at page 11, lines 1-26, and at Fig. 3, element 305.

Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004
Title: Detecting Incorrect Versions of Files

Docket: ROC920030419US1

Page 21

6. Grounds of Rejection to be Reviewed on Appeal

1. Whether claims 5-8 are non-statutory under 35 U.S.C. 101.
2. Whether claims 1 and 3-20 are non-statutory under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement.
3. Whether claims 1-4, 6-9, 11, and 14-20 are non-statutory under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

7. Argument

A) The Applicable Law

35 U.S.C. 112, first, second, and fifth paragraphs recites:

“The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

...

An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.”

MPEP 2111.01 (I) recites: “the words of the claim must be given their plain meaning unless the plain meaning is inconsistent with the specification. In re Zletz, 893 F.2d 319, 321, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989) (discussed below); Chef America, Inc. v. Lamb-Weston, Inc., 358 F.3d 1371, 1372, 69 USPQ2d 1857 (Fed. Cir. 2004).”

MPEP 2111.01 (III) recites:

“[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application.” Phillips v. AWH Corp., 415 F.3d 1303, 1313, 75 USPQ2d 1321, 1326 (Fed. Cir. 2005) (en banc). Sunrace Roots Enter. Co. v. SRAM Corp., 336 F.3d 1298, 1302, 67 USPQ2d 1438, 1441 (Fed. Cir. 2003); Brookhill-Wilk 1, LLC v. Intuitive Surgical, Inc., 334 F.3d 1294, 1298 67 USPQ2d 1132, 1136 (Fed. Cir. 2003) (“In the absence of an express intent to impart a novel meaning to the claim terms, the words are presumed to take on the ordinary and customary meanings attributed to them by

those of ordinary skill in the art."). It is the use of the words in the context of the written description and customarily by those skilled in the relevant art that accurately reflects both the "ordinary" and the "customary" meaning of the terms in the claims. Ferguson Beaugard/Logic Controls v. Mega Systems, 350 F.3d 1327, 1338, 69 USPQ2d 1001, 1009 (Fed. Cir. 2003) (Dictionary definitions were used to determine the ordinary and customary meaning of the words "normal" and "predetermine" to those skilled in the art.)"

35 U.S.C. 101 recites: "Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefore, subject to the conditions and requirements of this title."

MPEP 2106 (II) (C) recites:

"Where means plus function language is used to define the characteristics of a machine or manufacture invention, such language must be interpreted to read on only the structures or materials disclosed in the specification and "equivalents thereof" that correspond to the recited function. In re Donaldson, 16 F.3d 1189, 1193, 29 USPQ2d 1845, 1848 (Fed. Cir. 1994) (en banc); In re Alappat, 33 F.3d 1526, 1540, 31 USPQ2d 1545, 1554 (Fed. Cir. 1994) (en banc)."

B) Discussion of the Rejections

1. Claims 5-8 are rejected under 35 U.S.C. 101.

Claims 5-8

Claim 5-8 are rejected under 35 U.S.C. 101 because they "recite an 'apparatus' comprising a series of means that can be reasonable interpreted as software, per se." Appellant respectfully requests reversal of these rejections for the reasons argued below.

MPEP 2106 (II) (C) recites:

"Where means plus function language is used to define the characteristics of a machine or manufacture invention, such language must be interpreted to read on only the structures or materials disclosed in the specification and "equivalents thereof" that correspond to the recited function. In re Donaldson, 16 F.3d 1189, 1193, 29 USPQ2d 1845, 1848 (Fed. Cir. 1994) (en banc); In re Alappat, 33 F.3d 1526, 1540, 31 USPQ2d

1545, 1554 (Fed. Cir. 1994) (en banc).” (emphasis added).

Thus, since claims 5-8 recite an apparatus comprising various means plus functions, claims 5-8 must be interpreted to read on only the structures or materials disclosed in the specification and "equivalents thereof" that correspond to the recited function. Appellant's specification at page 5, line 17 through page 6, line 21 recites:

“The main memory 102 is a random-access semiconductor memory for storing data and programs. ... The memory 102 includes a debug controller 168 and a program 172. ... The debug controller 168 includes a classpath controller 170. ... In an embodiment, the classpath controller 170 includes instructions capable of executing on the processor 101 or statements capable of being interpreted by instructions executing on the processor 101 to access or communicate with the user interfaces as further described below with reference to Figs. 2, 3, and 4, and to perform the functions as further described below with reference to Figs. 5A and 5B. In another embodiment, the classpath controller 170 may be implemented in microcode. In yet another embodiment, the classpath controller 170 may be implemented in hardware via logic gates and/or other appropriate hardware techniques, in lieu of or in addition to a processor-based system.”

Thus, appellant's specification recites, e.g., the structure, materials, and physical components of a random-access semiconductor memory that stores instructions capable of executing on a processor, as a random-access semiconductor memory that stores statements capable of being interpreted by instructions capable of executing on a processor, and hardware implemented via logic gates, and claims 5-8 must be interpreted to read on only the structures or materials disclosed in the specification, so claims 5-8 are statutory under 35 U.S.C. 101.

2. Claims 1 and 3-20 are rejected under 35 U.S.C. 112, first paragraph.

Claims 1 and 3-20

Claims 1 and 3-20 are rejected under 35 U.S.C. 112 because “Applicant's specification does not adequately define what is meant by the terms ‘incorrect’ ... or how a

determination as to whether a particular file or class may be determined to be ‘incorrect.’” Appellant respectfully requests reversal of the Examiner’s rejections for the reasons argued below.

Appellant’s specification at page 3, lines 26-28 and page 4, lines 1-3 recites: “A method, apparatus, system, and signal-bearing medium are provided that in an embodiment issue a warning if a file to be used is an older version. In an embodiment, the warning includes an identification of the location of a newer version of the file. In an embodiment, the file is a class, and the old and new versions are found using a classpath, but in other embodiments any type of file or other object may be used. In this way, the use of incorrect versions of files may be detected.” (emphasis added).

Appellant’s specification at page 11, lines 25-26 recites: “The classpath controller 170 determines that the associated class may be the incorrect version as further described below with reference to Figs. 5A and 5B.”

Thus, blocks 505, 510, 515, 520, 525, 530, 555, 560, 565, and 575 of Figs. 5A and 5B, page 3, lines 26-28, page 4, lines 1-3, page 11, lines 25-26, and page 12, line 21 through page 14, line 16 of appellant’s specification define what is meant by the term “incorrect” and describe how a determination is made as to whether a particular file or class is “incorrect,” by way of example and not of limitation, so claims 1 and 3-20 are statutory under 35 U.S.C. 112.

Claims 1 and 3-20 are rejected under 35 U.S.C. 112 because “Applicant’s specification does not adequately define what is meant by the terms ... ‘older,’ or ‘newer’ ... or how a determination may be made as to whether a particular file or class may be determined to be ... ‘older’, or ‘newer.’” Appellant respectfully requests reversal of the rejections for the reasons argued below.

MPEP 2111.01 (I) recites:

“the words of the claim must be given their plain meaning unless the plain meaning is inconsistent with the specification. In re Zletz, 893 F.2d 319, 321, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989) (discussed below); Chef America, Inc. v. Lamb-Weston, Inc., 358 F.3d 1371, 1372, 69 USPQ2d 1857 (Fed. Cir. 2004).”

MPEP 2111.01 (III) recites:

“[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application.” Phillips v. AWH Corp., 415 F.3d 1303, 1313, 75 USPQ2d 1321, 1326 (Fed. Cir. 2005) (en banc). Sunrace Roots Enter. Co. v. SRAM Corp., 336 F.3d 1298, 1302, 67 USPQ2d 1438, 1441 (Fed. Cir. 2003); Brookhill-Wilk 1, LLC v. Intuitive Surgical, Inc., 334 F.3d 1294, 1298 67 USPQ2d 1132, 1136 (Fed. Cir. 2003) (“In the absence of an express intent to impart a novel meaning to the claim terms, the words are presumed to take on the ordinary and customary meanings attributed to them by those of ordinary skill in the art.”). It is the use of the words in the context of the written description and customarily by those skilled in the relevant art that accurately reflects both the “ordinary” and the “customary” meaning of the terms in the claims. Ferguson Beauregard/Logic Controls v. Mega Systems, 350 F.3d 1327, 1338, 69 USPQ2d 1001, 1009 (Fed. Cir. 2003) (Dictionary definitions were used to determine the ordinary and customary meaning of the words “normal” and “predetermine” to those skilled in the art.)”

Appellant uses the ordinary dictionary meaning of the terms “older” and “newer” and respectfully submits that determining whether one file is older or newer than another is a trivial exercise for a person of ordinary skill in the art, according the meaning of the terms “older” and “newer,” as described below in Evidence Appendix, Exhibit A, page 780 of the Merriam-Webster Collegiate Dictionary, 10th edition, 2000. Thus, claims 1 and 3-20 are statutory under 35 U.S.C. 112.

Claims 1 and 3-20 are rejected under 35 U.S.C. 112 because “while the specification illustrates ‘exemplary’ user interfaces ..., no description in the specification recites the necessary steps to acquire and display such information.” Appellant respectfully requests reversal of the rejections for the reasons argued below.

Appellant’s specification at page 8, line 26 through page 9, line 9 recites:

“The computer system 100 depicted in Fig. 1 has multiple attached terminals 121, 122, 123, and 124, such as might be typical of a multi-user “mainframe” computer system. Typically, in such a case the actual number of attached devices is greater than those shown in Fig. 1, although the present invention is not limited to systems of any particular size. The computer system 100 may alternatively be a single-user system, typically containing only a single user display and keyboard input, or might be a server or similar device which has little or no direct user interface, but receives requests from other computer systems (clients).”

Appellant’s specification at page 11, third full paragraph, lines 26-19 recites: “In response to the user selecting the icon 320, or in response to any other appropriate command or stimulus, the debug controller 168 obtains further information regarding the warning from the classpath controller 170 and displays the user interface of Fig. 4.”

Appellant’s specification at page 12, second full paragraph, lines 14-16 recites: “Although the notification 425 is illustrated as being a popup window, in other embodiments, the notification may be implemented as message, whether text or oral, or any other appropriate notification.”

Appellant respectfully submits that that implementation of user display devices, keyboards, popup windows, and selection of icons and the display and receipt of information using such items is well known to persons of ordinary skill in the art, as evidenced by Evidence Appendix Exhibit B, pages 323, 369, 686, and 745 of the IBM Dictionary of Computing, tenth edition, 1994, which describe an icon, a keyboard, a terminal, and a window, respectively. Thus, the claims are statutory under 35 U.S.C. 112.

Claims 1 and 3-20 are rejected under 35 U.S.C. 112 because “Applicant’s specification does not adequately define ... how a determination may be made as to whether such as user ‘owns’ the first/second file/class.” Appellant respectfully requests reversal of the rejections for the reasons argued below.

File and class ownership are well known to persons of ordinary skill in the art, as evidenced by Evidence Appendix Exhibit B, pages 104, 105, and 270 of the IBM Dictionary of Computing, tenth edition, 1994. Thus, the claims are statutory under 35 U.S.C. 112.

Claims 1 and 3-20 are rejected under 35 U.S.C. 112 because “Applicant’s specification does not adequately define what is meant by a user ‘doing debug.’” Appellant respectfully requests reversal of the rejections because the claims nowhere recite a user “doing debug.” Thus, the claims are statutory under 35 U.S.C. 112.

3. Claims 1-4, 6-9, 11, and 14-20 are rejected under 35 U.S.C. 112, second paragraph.

Claims 1-4

Claims 1-4 are rejected under 35 U.S.C. 112 because the term “incorrect” “is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.” Appellant respectfully submits that the claims are statutory under 35 U.S.C. 112 for the reasons argued below.

First, claim 1 recites: “determining whether the first file is an incorrect version, wherein the determining whether the first file is the incorrect version further comprises determining whether a second file later in the classpath from the first file is an earlier version than the first file,” which defines the term “incorrect.”

Second, appellant’s specification at page 11, lines 25-26 recites: “The classpath controller 170 determines that the associated class may be the incorrect version as further described below with reference to Figs. 5A and 5B.”

Thus, blocks 505, 510, 515, 520, 525, 530, 555, 560, 565, and 575 of Figs. 5A and 5B, and page 12, line 21 through page 14, line 16 of appellant’s specification provides a standard for ascertaining the requisite degree, and one of ordinary skill in the art would be reasonably apprised of the scope of the invention.

Thus, claims 1-4 are statutory under 35 U.S.C. 112.

Claims 17-20

Claims 17-20 are rejected under 35 U.S.C. 112 because “it is unclear what concrete steps are required by the various ‘configuring’ steps.” Appellant respectfully requests reversal of the rejections for the reasons argued below. Appellant respectfully submits that the claims are statutory under 35 U.S.C. 112 because appellant is using “configuring” with the

Appellant: Cary L. Bates

Page 29

Serial No.: 10/821,146

Filed: April 8, 2004

Docket: ROC920030419US1

Title: Detecting Incorrect Versions of Files

ordinary dictionary meaning, which is well known to persons of ordinary skill in the art, as evidenced by Evidence Exhibit A, page 241 of the Merriam-Webster Collegiate Dictionary, 10th edition, 2000.

Claims 6-9, 11, and 14-16

Claims 6-9, 11, and 14-16 are statutory under 35 U.S.C. 112 because they do not contain “incorrect” or “configuring.”

Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004
Title: Detecting Incorrect Versions of Files

Page 30

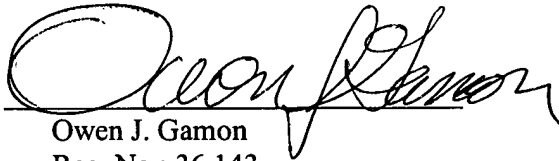
Conclusion

Appellant respectfully requests reversal of the above rejections. If the Board is of the opinion that any rejected claim may be allowable in amended form, then appellant also respectfully requests a statement to that effect.

Respectfully submitted,

Date June 3, 2008

By



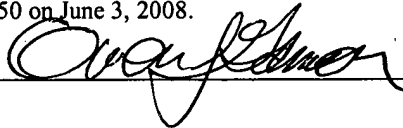
Owen J. Gamon
Reg. No.: 36,143
phone: 651-645-7135
fax: 651-457-5622

IBM Corporation
Intellectual Property Law
Dept. 917, Bldg. 006-1
3605 Highway 52 North
Rochester, MN 55901

Certificate under 37 C.F.R. 1.8: I hereby certify that this correspondence is being transmitted via facsimile (Fax no: 571-273-8300) to the Commissioner of Patents, or is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner of Patents, Alexandria, VA 22313-1450 on June 3, 2008.

Name Owen J. Gamon

Signature



8. CLAIMS APPENDIX

1. A method comprising:

finding a first file in a first directory specified in a classpath;
determining whether the first file is an incorrect version, wherein the determining whether the first file is the incorrect version further comprises determining whether a second file later in the classpath from the first file is an earlier version than the first file; and
if the first file is the incorrect version, issuing a warning.

3. The method of claim 1, wherein the issuing further comprises:

providing an identification of a location of a newer version of the first file.

4. The method of claim 1, wherein the determining further comprises:

determining whether a second file is owned by a user and the first file is not owned by the user, wherein the second file is later in the classpath than the first file.

5. An apparatus comprising:

means for finding a first class in a first directory specified in a classpath;
means for finding a second class in a second directory, wherein the second directory is later in the classpath than the first directory; and
means for determining whether the second class is a newer version of the first class.

6. The apparatus of claim 5, further comprising:

means for issuing a warning if the second class is the newer version of the first class.

7. The apparatus of claim 5, further comprising:

mean for deciding whether the second class is owned by a user and the first class is not owned by the user.

8. The apparatus of claim 7, further comprising:

means for issuing a warning if the second class is owned by the user and the first class is not owned by the user.

9. A storage medium encoded with instructions, wherein the instructions when executed comprise:

finding a first class in a first directory specified in a classpath;

finding a second class in a second directory, wherein the second directory is later in the classpath than the first directory;

determining whether the second class is a newer version of the first class; and

issuing a warning if the second class is the newer version of the first class .

10. The storage medium of claim 9, further comprising:

deciding whether the second class is owned by a user and the first class is not owned by the user.

11. The storage medium of claim 10, further comprising:

issuing the warning if the second class is owned by the user and the first class is not owned by the user.

12. The storage medium of claim 9, further comprising:

saving a reason for the warning.

13. A computer system comprising:

a processor; and

memory encoded with instructions, wherein the instructions when executed on the processor comprise:

finding a first class in a first directory specified in a classpath,

finding a second class in a second directory, wherein the second directory is later in the classpath than the first directory, and

deciding whether the second class is owned by a user and the first class is not owned by the user.

14. The computer system of claim 13, wherein the instructions further comprise:

issuing a warning if the second class is owned by the user and the first class is not owned by the user.

15. The computer system of claim 14, wherein the issuing further comprises:

providing an identification of the second directory.

16. The computer system of claim 13, wherein the instructions further comprise:

determining whether the second class is a newer version of the first class; and
issuing a warning if the second class is the newer version of the first class.

17. A method of configuring a computer, wherein the method comprises:

configuring the computer to find a file in a first directory specified in a classpath;
configuring the computer to determine whether the file is an older version; and
configuring the computer to issue a warning if the file is the older version.

18. The method of claim 17, further comprising:

configuring the computer to search for a newer version of the file later in the classpath
from the older version.

19. The method of claim 17, wherein the warning further comprises:

an identification of a location of a newer version of the file.

20. The method of claim 17, wherein the file comprises a class.

Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004 Docket: ROC920030419US1
Title: Detecting Incorrect Versions of Files

Page 35

9. EVIDENCE APPENDIX

Attached are Exhibits A and B.

**Merria
Col
Dic**
TEN

THE WORDS!

acid snow;
La Niña
thermog;

New entries lil
Merriam-We
every year, contri
100 years ago of
words and meani

EASY-TO-UNI
Communicate el
than 215,000 cl

AUTHORITAT
Discover the cc
as where to hy
(including vari
how long it's t

USAGE GUI
SYNONYM I
Choose the ri
guidance prov
and usage gu

UNMATCH
Rely on the l
Webster, put
New Interna
America's n
of the Engli

Merriam-

deli:



A GENUINE MERRIAM-WEBSTER

The name *Webster* alone is no guarantee of excellence. It is used by a number of publishers and may serve mainly to mislead an unwary buyer.

Merriam-Webster™ is the name you should look for when you consider the purchase of dictionaries or other fine reference books. It carries the reputation of a company that has been publishing since 1831 and is your assurance of quality and authority.

Copyright © 2000 by Merriam-Webster, Incorporated

Philippines Copyright 2000 by Merriam-Webster, Incorporated

Library of Congress Cataloging in Publication Data
Main entry under title:

Merriam-Webster's collegiate dictionary. — 10th ed.
p. cm.

Includes index.

ISBN 0-87779-708-0 (unindexed : alk. paper). — ISBN 0-87779-709-9
(indexed : alk. paper). — ISBN 0-87779-710-2 (deluxe indexed : alk. paper).
— ISBN 0-87779-707-2 (laminated cover, unindexed).

1. English language—Dictionaries. I. Merriam-Webster, Inc.

PE1628.M36 1998
423—dc21

97-41846
CIP

Merriam-Webster's Collegiate® Dictionary, Tenth Edition principal copyright 1993

COLLEGIATE is a registered trademark of Merriam-Webster, Incorporated

All rights reserved. No part of this book covered by the copyrights hereon may be reproduced or copied in any form or by any means—graphic, electronic, or mechanical, including photocopying, taping, or information storage and retrieval systems—without written permission of the publisher.

Made in the United States of America

2728293031RT:WC00

²**newborn** *n, pl newborn or newborns* (1879)
New-burg or New-burgh \nū-,bɜrg, 'ny-

(1902) : served with a sauce made of cream and egg yolks (lobster ~) (shrimp ~)

Newcastle disease ('nu-,ka-sal-, nyu-) [*Avian influenza* (Tyne, England) (1927)]: a destructive virus disease of domestic fowl that is caused by a paramyxovirus.

new-com-er 'nū-,kə-mər, 'nyū-\ n (15c)-
: BEGINNER

New Criticism *n* (1941): an analytic literary criticism by concentration on the language, imagery, and structural tensions in literary works — New Criticism

New Deal *n* [fr. the supposed resemblance] and equality of opportunity afforded by

(1932) : the legislative and administrative
Roosevelt designed to promote economic
during the 1930s; *also* : the period of this pro

New Deal-ish \-'dē-lish\ *adj* — New Deal new drug *n* (ca. 1951): a drug that has not been tested by qualified experts under the conditions

ive by qualified experts under the conditions
ed, or suggested in the label and that may
or an established drug prescribed for use in

new economics *n pl* but usu sing in constr (US) that is a logical extension of Keynesianism; appropriate fiscal and monetary maneuvering can

new-el \ 'nū-əl, 'nyū-əl *n* [ME *nowell*, fr. MF *nuovo* like a put, fr. L *nuc-* *nux* *nut* =

nucalis like a nut, fr. L *nuc-*, *nux*-nut — *mo* right post about which the steps of a circular at the foot of a straight stairway or one of a

New English Bible *n* (1957) : a translation of the Bible by an interdenominational committee first published in 1961
new-fan-gled \nū-'fan-gəld\ *adj* [ME *nyu* *fan* *gled*]

sumed) OE *-fangol*, fr. OE *fōn* (pp. *fangan* PACT) (14c) 1: attracted to novelty. 2: had many gadgets in the kitchen.

new-fash-ioned \-'fa-shənd\ *adj* (f-592)
form 2 : UP-TO-DATE

new-found \-'faʊnd\ *adj* (15c) : newly found
New-found-land \'nū-fən(d)-lənd; 'ny-
 [Newfoundland, Canada] (1773) : any of the

highly intelligent black, black and white, or
Newfoundland

New Greek *n* (ca. 1958) : Greek as used by the medieval period

New Hebrew *n* (ca. 1959) : the Hebrew language in Israel

sizes the historicity of a text by relating it to its social, political, economic, or ideological context. The new *le 'lūnā* 'nvil' 'n (ca. 1945) is similar to

New Jer-sey tea \nù-'jər-zē- nyu-'jər-zē/ *Nerium*
use of its leaves as a substitute for tea during

(1759) : a low deciduous shrub. (Geophila) thorn family that is found in the eastern and small white flowers borne in large cl

New Je·ru·sa·lem \-jə-'rū-s(ə)-ləm, -zə-ləm
 "the holy city, *New Jerusalem*" (Rev.21:2)
 of souls redeemed by Christ 2: an ideal

New Journalism n (1967): journalism with subjective responses to people and events and

New Latin *n* (ca. 1889) : *Latina* used since the
 period esp. in scientific description and official

New Left *n* (1960): a political movement of students in the 1960s, favoring confrontation with older leftist ideologies and concerned

new-ly \ 'nū-lē, 'nyū- \ *adv.* (before 12c) *all* the

new-mar-ket \ 'nū-,mār-kēt, \ (nyū-'mār-kēt)

long close-fitting coat worn in the 19th century.
new math *n* (1964) : basic mathematical abstraction and the principles of set theory.

new moon *n* (bef. 12c) 1: the moon's phase with the sun so that its dark side is toward earth

with the sun so that its dark side is toward the crescent moon seen shortly after twilight. The occurrence of the new moon phase is the

New Right n (1966): a political movement opposed esp. to secular humanism and co.

opposed esp. to secular humanism, church and state, patriotism, race, fair and abortion

news \ˈnuːz, ˈnyuːz\ *n pl* but sing in constr. of recent events **b**: previously unknown to you **2 a**: material reported in a newspaper

news agency *n* (1873): an organization

news-agent \ 'nūz-ā-jent, -hūz- / (18)

news-boy \-,bɔɪ\ *n* (1764): one who sells newspapers
news-break \-,brāk\ *n* (1944): a report of a startling event
news-cast \-,kɑst\ *n* (1944): a broadcast of news

news-cast \-,kast/ *n* (news + broadcast)
broadcast of news — **news-caster** *n*
news conference *n* (1946) (PRISM CONFERENCE)

news-deal-er \-dē-lər/ *n.* (1861) a dealer in
and often paperback books.
news-group \-gru:p/ *n.* (1993) a group of news

ternet that is devoted to a particular topic.

Limitation of Liability

While the Editor and Publisher of this book have made reasonable efforts to ensure the accuracy and timeliness of the information contained herein, neither the Editor nor the Publisher shall have any liability with respect to loss or damage caused or alleged to be caused by reliance on any information contained herein.

Copyright © 1994 by International Business Machines Corporation. All rights reserved. Printed in the United States of America. Except as permitted under the United States Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a data base or retrieval system, without the prior written permission of the publisher.

7 8 9 0 DOC/DOC 9 9 8

ISBN 0-07-031488-8 (HC)

ISBN 0-07-031489-6 (PBK)

The sponsoring editor for this book was Daniel A. Gonneau and the production supervisor was Thomas G. Kowalczyk.

Printed and bound by R. R. Donnelley & Sons Company.

Tenth Edition (August 1993)

This is a major revision of the *IBM Dictionary of Computing*, SC20-1699-8, which is made obsolete by this edition. Changes are made periodically to the information provided herein.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming, or services in your country. Comments may be addressed to IBM Corporation, Department E37/656, P. O. Box 12195, Research Triangle Park, NC 27709.

International Edition

Copyright © 1994 by International Business Machines Corporation. Exclusive rights by McGraw-Hill, Inc. for manufacture and export. This book cannot be re-exported from the country to which it is consigned by McGraw-Hill. The International Edition is not available in North America.

When ordering this title, use ISBN 0-07-113383-6.

This book is printed on acid-free paper.

CIOCS

[104]

class

CIOCS Communications input/output control system.

cipher block chain (CFB) mode A mode of operation that cryptographically connects one block of ciphertext to the next plaintext block.

cipher feedback A cryptographic operation that exclusively ORs the plaintext with the output of a cryptographic algorithm and feeds the result back as input to the cryptographic algorithm. See also output feedback.

cipher system In computer security, any system of communications in which groups of symbols are used to represent plaintext elements of equal length. Contrast with code system. See also cryptographic system.

ciphertext (1) In computer security, text produced by encryption. (2) Synonym for enciphered data.

circuit (1) One or more conductors through which an electric current can flow. See physical circuit, virtual circuit. (2) A logic device.

circuit breaker A switch that automatically interrupts an electric circuit because of an abnormal condition.

circuit grade The information-carrying capability of a circuit, expressed in speed or type of signal. The grades of circuits are broadband, voice, subvoice, and telegraph. For data use, these grades are identified with certain speed ranges.

circuit load Synonym for line load.

circuit noise level The ratio of circuit noise to some arbitrary amount chosen as reference. This ratio is normally indicated in decibels above the reference noise (dbn) or in adjusted decibels (dba) that signify a noise meter reading adjusted to represent the interfering effect under specified conditions. See also decibel.

circuit-switched connection A connection that is established and maintained on demand between two or more data stations in order to allow the exclusive use of a data circuit until the connection is released. (T)

circuit-switched data transmission service A service using circuit switching to establish and maintain a connection before data can be transferred between data terminal equipment (DTEs). (T) See also packet switched data transmission service.

circuit switching (1) A process that, on demand, connects two or more data terminal equipment (DTEs) and permits the exclusive use of a data circuit between them until the connection is released. (I) (A)

(2) Synonymous with line switching. (3) See also message switching, packet switching.

circular shift Synonym for end-around shift.

circulating register A shift register in which data moved out of one end of the register are reentered into the other end as in a closed loop. (A)

circulating storage Dynamic storage involving a closed loop. (A) Synonymous with cyclic storage.

CKD (1) Count-key-data. (2) Cryptographic key distribution center.

CKDS In the Programmed Cryptographic Facility, cryptographic key data set.

CKT Cryptographic key translation center.

CL Control language.

cl Centiliter.

CLA Communication line adapter.

cladding In an optical cable, the region of low refractive index surrounding the core. See also core, optical fiber.

clamp-on Synonym for camp-on.

C language A language used to develop software applications in compact, efficient code that can be run on different types of computers with minimal change.

Clark-Wilson Integrity model See computer security model.

class (1) In object-oriented design or programming, a group of objects that share a common definition and that therefore share common properties, operations, and behavior. Members of the group are called instances of the class. (2) In RACF, a collection of defined entities (users, groups, and resources) with similar characteristics. The class names are USER, GROUP, DATASET, and the classes that are defined in the class descriptor table. (3) Any category in which things are assigned or defined. (4) In VSE/POWER, a means of grouping jobs that require the same set of resources. (5) In System/38, an object that contains the execution parameters for a routing step. (6) In the AIX Base Operating System, pertaining to the I/O characteristics of a device. System devices are classified as block or character devices. (7) In AIX Enhanced X Windows, a general group in which a specific object belongs. (8) In the AIXwindows program, an object-oriented data structure containing generalized information about a group of similar graphical objects known as widgets.

class of graphical objects inherits some appearance characteristics and behavior from the classes that precede it in the hierarchy.

class user Any user, regardless of class, who is authorized to execute a subset of CP commands to log onto the system, or a terminal logically connected to a virtual machine, or send messages to other users.

class values In the AS/400 system, the values in a class control the processing of routing steps. The values include the run priority, time of day, purge, default wait time, maximum temporary storage, and maximum temporary storage.

class primary system operator privilege

class authority (CLAUTH) In RACF, an authority used to define RACF profiles in a class descriptor table. A user can have authority to one or more classes.

class system resource operator privilege

class name In COBOL, the proposition, for which a truth value can be determined, that the contents of a data item are wholly alphabetic or is wholly numeric or contains only those characters listed in the definition of a class-name.

class system programmer privilege class.

class RACF-supplied control block for the class descriptor table, which are defined except the USER, GROUP, and DATASET classes.

class control table (CDT) In RACF, a table containing one entry for each class except the USER, GROUP, and DATASET classes. The table is generated by the ICHERCDE macro once for each class.

class spooling operator privilege class.

class system analyst privilege class.

class service representative privilege class.

class general user privilege class.

class transition In computer security, the determination of the degree of information requires a specific degree of security. (2) In computer security, a designation signifying the result of such a determination; for example, "top secret, secret, confidential." See also category, compartment.

class lock In the IBM 8100 Information System, a set of locks, including sublocks, used to lock a member of a class of resources.

class method In System Object Model, an action that can be performed on a class object. Synonymous with factory method.

class-name In COBOL, a user-defined word defined in the SPECIAL-NAMES paragraph of the Environment Division that assigns a name to the proposition, for which a truth value can be defined, that the contents of a data item consists exclusively of those characters listed in the definition of a class-name.

class object In the AS/400 system, an object that identifies the run attributes of a job.

class (of entities) In a conceptual schema language, all possible entities in the universe of discourse for which a given proposition holds. (A)

class of service (COS) (1) A set of characteristics (such as route security, transmission priority, and bandwidth) used to construct a route between session partners. The class of service is derived from a mode name specified by the initiator of a session. (2) See also user class of service.

class of service database A database maintained independently by each network node, and optionally by APPN end nodes. It contains one entry per class-of-service name; each database entry contains:

- A definition of the acceptable values for transmission group (TG) and node characteristics for routes described by that class-of-service name and the weight function to be used to compute the weights of nodes and TGs that meet the acceptable values
- The transmission priority to be used for traffic that flows on routes described by that class-of-service name.

class-of-service description In the AS/400 system, a system object created for an Advanced Peer-to-Peer Networking (APPN) node that provides the information required to assign relative priority to the transmission groups and intermediate routing nodes for an APPN session.

class transition In the IBM ImagePlus system, a change in an object's management class and/or storage class when an event occurs that brings about a change in an object's service level or management criteria.

file gap

file gap An area on a data medium intended to be used to indicate the end of a file and, possibly, the start of another. A file gap is frequently used for other purposes, in particular, as a flag to indicate the end or beginning of some other group of data. (A)

file ID In Subsystem Support Services, two alphanumeric characters that identify the kind of record a data set entry contains.

file index Synonym for i-node.

file key In RPG, all the key fields defined for a file.

file layout The arrangement and structure of data or words in a file, including the order and size of the components of the file. (I) (A)

file level specifications In the AS/400 system and System/38, specifications coded on the lines before the first record format name. See also field level specifications, key field level specifications, help level specifications, join level specifications, record level specifications, select/omit level specifications.

file lock In the AIX operating system, a means to limit or deny access to a file by other users. A file lock can be a read lock or a write lock.

file maintenance (1) The activity of updating or reorganizing a file. (T) (2) Adding, changing, or deleting records in a file to keep the information in the file current.

file management (1) Creation and maintenance of files by means of a computer. (2) In personal computers, the use of application software to access, create, modify, store, and retrieve files and to obtain documents such as reports and mailing lists.

file name (1) A name assigned or declared for a file. (2) The name used by a program to identify a file. See also label.

file-name In COBOL, a user-defined word that names a file connector described in a file description entry or a sort-merge file description entry within the File Section of the Data Division.

file name substitution In the AIX operating system, the process in which the shell recognizing a word (character string) that contains any of the *, ?, [, or { characters, or begins with the ~ character, and replaces it with an alphabetically sorted list of file names that match the pattern of the word. Synonymous with globbing.

file operation code In RPG, an operation code, such as CHAIN, that lets the user control the input/output operations to a file.

[270]**file recovery**

file organization (1) The physical order of the stored records that comprise the contents of a particular file and that determines the access method that must be implemented in order to provide for entry into a database. (T) (2) In COBOL, the permanent logical file structure established at the time that a file is created. See indexed organization, relative organization, sequential organization.

file overrides The file attributes specified at execution time that will override the attributes specified in the file description or in the program.

file owner In the AIX operating system, the user who has the highest level of access authority to a file, as defined by the file.

file pointer (1) In the AIX operating system, an identifier that indicates a structure containing the file name. (2) In Pascal, an identifier that indicates the location of an item of data in an input/output buffer.

file position indicator In COBOL, a conceptual entity that contains the value of the current key within the key of reference for an indexed file, or the record number of the current record for a sequential file, or the relative record number of the current record for a relative file, or that indicates that no next logical record exists, or that the number of significant digits in the relative record number is larger than the size of the relative key data item, or that an optional input file is not present, or that the at end condition already exists, or that no valid next record has been established.

file profile In RACF, a description of a RACF-defined file, including file name, owner, universal access authority, security level, and other data.

file protected Pertaining to a tape reel with the write-enable ring removed.

file protection In computer security, the processes and procedures established in an information system that are designed to inhibit unauthorized access to, contamination of, or deletion of a file.

file-protection ring A removable plastic or metal ring on a magnetic tape reel, the presence or absence of which prevents writing on the magnetic tape and thereby prevents the accidental erasure of a file. Synonymous with safety ring, file protect ring. (T)

file protect ring Synonym for file-protection ring. (T) (A)

file recovery See backward file recovery, forward file recovery.

file referen

file referen
ceded by a
file in an

file referen
and System
user track fi

File Section
sion that co
their associa

file selectio
choose a fi
from the on
space provic

file separat
beginning o
the other fil

file separat
rator intende
files. (I) (

file server
computer th
access and r
attached cor
sonal Comp
See also disl

Note: Acce
server's soft
of the comp

Fi
Ser

Figure 62. f

files library
that contains
jobs.

file spec File

10/821,146

Page 43

[323]

identifier

products

mail and
PROBS
Comm
e.

ng to the
the oper
The OS/2
sks at the

it-end data
OS/2 oper
interface

(PSR) An
maintained
lized IBM
neer IBM

that can be
vert it to a

ndard PS/2
if receiving
Committee
of sources
ana, video
er.

unications
rates on a
X, or 43V
mands, and
ers; remote
ch systems
See also

ual in IBM
IM products
rt Represent

IBM depart

pecialist An
maintained
ee also IBM
rt Represent

multimedia

IC Integrated circuit.

ICA (1) International Communication Association, formerly called Industrial Communication Association.
(2) Integrated communication adapter.

ICB Interrupt control block.

ICCF Interactive computing and control facility. See ICE/CCF.

ICF Intersystem communications function.

ICF file A device file that allows a program on one system to communicate with a program on another system. There can be one or more sessions with the same or different communications devices at the same time.

ICM memory Integrated circuit memory. (A)

ICMP Internet Control Message Protocol.

ICNCB Intelligent controller node control block.

icon (1) A graphic symbol, displayed on a screen, that a user can point to with a device such as a mouse in order to select a particular function or software application. Synonymous with pictogram. (T)
(2) In SAA Advanced Common User Access architecture, a graphical representation of an object, consisting of an image, image background, and a label.

icon box In the AIXwindows program, a window used as a visual storage area for icons representing minimized windows.

icon layout policy In the AIXwindows program, a specification that determines whether icons representing minimized windows are placed on the root window or within an icon box.

ICR Independent component release.

ICU Interactive chart utility.

ICV Initial chaining value.

ICW (1) Initial control word. (2) Interface control word.

ID (1) Identifier. (2) Identification.

IDU Interactive data definition utility.

idea processor Personal computer application software that allows a user to organize thoughts in outline form and modify, expand, compress, and reorganize topics as required.

identification In computer security, the process that allows a system to recognize an entity by means of personal, equipment, or organizational characteristics or codes.

identification card In the 3600 Finance Communication System, a card similar to a credit card that contains a customer's identification number written on a magnetic stripe. Customers insert the identification card in the 3614 Consumer Transaction Facility to identify themselves. See also personal code.

identification card reader In the 3614 Consumer Transaction Facility, a component that reads precoded information from the magnetic stripe on a customer's identification card.

Identification Division One of the four main parts of a COBOL program. In addition to identifying the source program and the object program, this part may also describe the author's name, the location where written, and the date written.

identification (ID) characters Characters sent by a station on a switched line to identify the station. TWX, BSC, and SDLC stations use ID characters.

identification number See customer identification number.

identifier (1) One or more characters used to identify or name a data element and possibly to indicate certain properties of that data element. (A) (2) In programming languages, a token that names a data object such as a variable, an array, a record, a subprogram, or a function. (A) (3) In the C language, a sequence of letters, digits, and underscores used to identify a data object or function. (4) In COBOL, a syntactically correct combination of a data name, with its qualifiers, subscripts, and reference modifiers, as required for uniqueness of reference, that names a data item. The rules for an identifier associated with the general formats may, however, specifically prohibit qualification, subscripting, or reference modification. See resultant identifier. (5) In FORTRAN, a lexical unit that names a language object; for example, the names of variables, arrays, and program units. The name of a declared unit. (6) In Pascal, a lexical unit that names a language object; for example, the names of variables, arrays, records, labels, and procedures. The name of a declared item. (7) In PL/I, a single alphabetic character or a string of alphabetic characters, digits, and break characters that starts with an alphabetic character. identifier, ordinary identifier. (8) A sequence of bits or characters that identifies a program, device, or system to another program, device, or system. (9) In the AIX Enhanced X-Windows program, a unique value associated with a resource that a client program uses to name the resource. An identifier can be used over any con-

Kerning

[369]

key compression

hat has a total string increment. Toned by increment on either

more, kerned character

ing system that can allocating hardware

) A program that can system environment program that must be ad other parts of the initial program and also security kernel ing system for RISC is that are needed are

ing system, symbol in

ing system, the state l mode. Contrast with

operating system, kernel allocates certain s with system param

space between two ple, overlapping characters so that their on picture elements (pels) he character cell

r boxes to overlap and that characters can be ligatures, or any other more than one character of proportional spaced r boxes, characters can y can be placed farther ; character boxes. See

Ty

With Kerning

(1) An identifier within a set of data elements. (T) (2) One or more characters used to identify the record and establish the order of the record within an indexed file. (3) In VSAM, one or more consecutive characters taken from a data record, used to identify the record and establish its order with respect to other records. See also alternate key, key table. (4) In ACF/TCAM, a character string that matches a definition in the key table. This key identifies the destination of a message or special processing to be done on that message. See also key table. (5) Identifier information from a keyboard. (6) In computing, synonym for control word. (7) In computer security, a sequence of symbols used with a cryptographic algorithm for encrypting or decrypting data. See: key-encrypting key, key-exchange key, master key, private key, public key. (8) The value used to identify a record in a keyed sequence file. (9) In SQL, a column or an ordered collection of columns identified in the description of an index. (10) In COBOL, a data item that identifies the location of a record, or a set of data items that serve to identify the ordering of data.

(11) authentication code A key test pattern.

Keyboard: (1) An arrangement of typing and function keys laid out in a specified manner. (T) (2) A systematic arrangement of keys by which a machine is operated or by which data are entered. (3) A device used to encode data by key depression, which causes generation of the selected code element. (4) A group of numeric keys, alphabetic keys, or function keys used for entering information into a terminal and into the system. (5) See also AZERTY, DVORAK, QWERTY. See Figure 82.

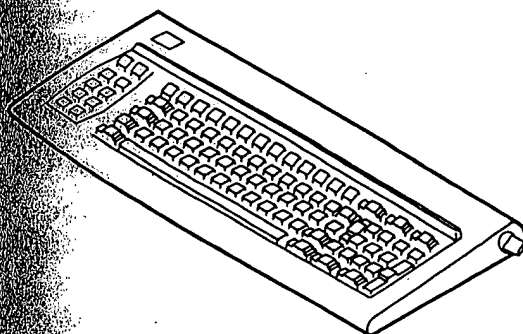


Figure 82. Keyboard

Keyboard grabbing In the AIX Enhanced Windows program, the process by which a client can actively take control of the keyboard; key events are sent to that client, rather than the client to which they would normally be sent. See also active grab, button grabbing, key grabbing, passive grab, pointer grabbing, server grabbing.

keyboard mapping A list, usually in a profile, that establishes correspondence between each key on the keyboard and the character displayed on a display screen, or an action taken by a program, when that key is pressed.

keyboard number An identification number that appears on certain keyboards that are used outside of the United States to help the user identify the keyboard layout.

keyboard overlay A template placed on a keyboard to explain the function of each key for a particular application program or for a keyboard program.

keyboard program Personal computer application software that changes key functions; for example, from a QWERTY layout to an AZERTY layout.

keyboard punch Synonym for keypunch.

keyboard send/receive (KSR) A combination teletypewriter transmitter and receiver with transmission capability from keyboard only.

keyboard send-receive (KSR) mode In the AIX operating system, a mode in which a virtual terminal emulates a standard ASCII terminal during input and output operations.

keyboard shift In AS/400 Data Description Specifications (DDS), a characteristic that can be specified for a field in a display file that automatically shifts the display station keyboard to control what the display station user can enter into the field. In the interactive data definition utility (IDDU) and (DDS) the keyboard shift can also be specified in database files, but only applies when these fields are referred to in a display file.

keyboard type The physical key arrangement and assignments for a keyboard.

key click Transient pulses or surges on a transmission line set up by the opening or closing of keying circuit contacts.

key-click filter A filter that attenuates key clicks.

key component (1) In cryptography, one of at least two characters having the format of a cryptographic key that is exclusive-ORed with one or more like parameters to form a cryptographic key. (2) In cryptography, any one of a combination of elements that make up a cryptographic key.

key compression The elimination of characters from the front and back of a key that VSAM does not need to distinguish the key from the preceding or following

temporary objects

[686]

terminal

temporary objects In the AS/400 system, objects, such as data paths or compiler work areas, that are automatically deleted by the system when the operating system is loaded.

temporary read/write error An error that is eliminated by retrying a read/write operation.

temporary storage In computer programming, storage locations reserved for intermediate results. (A) Synonymous with working storage.

temporary text delay (TTD) A control character sequence sent by a transmitting station either to indicate a delay in transmission or to initiate an abort of the transmission in progress.

temporary-text-delay (TTD) character In BSC, a transmission control character that is used to maintain the data link when no text is being transmitted. TTD indicates to the receiving station that there is a temporary delay in transmission of data.

tens complement The radix complement in the decimal numeration system. (I) (A) Synonymous with complement-on-ten.

tensile strength A measure of the tension that a material such as continuous forms can accept without tearing.

tensioning In the 3800 Printing Subsystem, stretching or causing extension of continuous forms while they are threaded in the printer.

tera (1) Ten to the twelfth power; 1,000,000,000,000 in decimal notation. When referring to storage capacity, two to the fortieth power; 1,099,511,627,776 in decimal notation. (2) A unit of measure equal to 10^{12} bytes; 1 000 000 000 000 in decimal notation.

term (1) A construct in a conceptual schema language that refers to an entity. (T) (2) The smallest part of an expression that can be assigned a value. (3) See absolute term, arithmetic term, logical term, relocatable term.

terminal (1) A functional unit in a system or communication network at which data may enter or leave. (T) (2) A point in a system or communication network at which data can either enter or leave. (A) (3) A device, usually equipped with a keyboard and display device, capable of sending and receiving information. (4) In COBOL, the originator of a transmission to a queue, or the receiver of a transmission from a queue.

Note: The terms terminal and workstation are often used interchangeably. However, a terminal may not

have a human operator. A workstation is a terminal at which a human operator performs an application.

terminal access facility (TAF) In a network program, a facility that allows a network operator to control a number of subsystems. In a terminal operator control session, operators can control a combination of such subsystems simultaneously.

terminal address card In a 3600 Final Communication System, an addressable logic element that connects a terminal to a local loop or to a loop adapter that handles signals passing through the loop adapter. Synonymous with terminal loop adapter.

terminal component A separately addressable component of a terminal that performs an input or output function, such as the display component of a terminal or a device or a printer component of a terminal.

terminal configuration facility (TCF) A facility that allows macrostatements to be coded by the user to define the environment in programmable store system host environment used to define and create the terminal environment.

terminal control address space (TCAS) A TSO/VTAM that provides logical address space for TSO/VTAM users.

Terminal Control table (TCT) A table that contains the configuration of terminals, logical units, and other systems in a CICS network with which the terminal system communicates.

terminal descriptor In the AIX object model (ODM), a named variable of the type binary, char, or vchar used to define the types in an ODM object class definition. (A) Synonymous with binary, char, long, object class, short, vchar.

terminal display language (TDL) A language for coding statements by the application programmer to control the IBM 3275 Display Station. TDL statements define the formats of data sent between a keyboard/display and the application program buffer. They require translation by a terminal definition language translator. Synonymous with information definition language.

terminal emulation The capability of a computer or personal computer to operate as a particular type of terminal linked to a host system and to access data. See also download, upload.

terminal emulator A program that allows a computer, such as a microcomputer or personal computer, to send and receive data from a computer system.

get gravity

ed the same

nt copying
quid devel-
opment.

formance of

g statement
wed by an
and a state-

ig a video
s are to be
that colors

r.

y.

rk that pro-
graphic area
etwork or a
nay use or
(T) (2) A
to serve an
for example,
tworks, and
th local area

i, a graphic
keyboard or
tion or with
every widget
ways has a
amental data
lowsToolkit.
ser-interface
get. It is the
ows window
antics. A
widget class

, the general
gs. It is a
idget type.

ity.

widget ID

[745]

window gravity

widget ID In the AIX operating system, a unique identification number associated with each widget instantiated in an interface.

widget instance In the AIX operating system, a specific widget object, as opposed to a general widget class. It is composed of a data structure containing instance-specific values and another data structure containing information applicable to all widgets of that class.

widget tree (1) In the AIX operating system, the symbolic structure for Enhanced X-Windows Toolkit code. The basic element is a widget class. See also leaves, intermediate nodes, root. (2) A hierarchy of widgets within a specific client application. The Shell widget is the root of the widget tree. Widgets with no children of any kind are leaves of the tree.

widget type Synonym for widget class.

widow (1) A last line of a paragraph that is carried over to the top of the next column or page, where it stands alone. Synonymous with widow line. (T) (2) In word processing and desktop publishing, a heading, a line, or a few lines of text beginning a paragraph that are printed or displayed at the end of a page. See also orphan.

widow line Synonym for widow. (T)

width slots In the 3800 Printing Subsystem, the openings that mechanically lock the paper width lever in the continuous forms stacker. A specific opening is associated with each of the paper widths.

wildcard character Synonym for pattern-matching character.

wild footage Synonym for original footage.

willful intercept The act of intercepting messages intended for stations having equipment or line trouble. See also miscellaneous intercept.

Winchester Pertaining to a technology used in hard disk drives in which a movable read/write head floats above a rotating disk on a cushion of air produced by the rotating surface. When rotation stops, the air cushion is lost and the head comes to rest on the surface of the disk in a landing zone where no data recorded. See also Bernoulli, landing zone, loading zone.

Note: Winchester drives are sealed to prevent contamination that can cause head crashes.

window (1) A portion of a display surface in which display images pertaining to a particular application can be presented. Different applications can be dis-

played simultaneously in different windows. (A) (2) An area of the screen with visible boundaries within which information is displayed. A window can be smaller than or the same size as the screen. Windows can appear to overlap on the screen. (3) A division of a screen in which one of several programs being executed concurrently can display information. (4) In data communication, the number of data packets a DTE or DCE can send across a logical channel before waiting for authorization to send another data packet. The window is the main mechanism of pacing, or flow control, of packets. (5) In MSS, the portion of a sequential data set on a virtual volume that can be staged for processing. A window is a multiple of a page (8 cylinders) and can range from 2 to 25 pages. (6) In the AIX operating system, a rectangular area of the screen that a user can move about, place on top of another window, pull under another window, or iconize. (7) In AIX curses and extended curses, the internal representation of what a portion of the display may look like at some point in time. Windows can be any size, from the entire display screen to a single character.

Window (1) In SAA Advanced Common User Access architecture, a choice in the action bar of some applications. Users select it to arrange the display of several windows or to change the active window. (2) In SAA Advanced Common User Access architecture, a choice in the action bar of multiple-document interface applications. (3) In SAA Advanced Common User Access architecture, an icon that represents a general window in a user's work or parts box.

window class The grouping of windows whose processing requirements conform to the services provided by one window procedure.

window class style The set of properties that apply to every window in a window class.

window component In SAA Advanced Common User Access architecture, the smallest named visual part of a window, such as the title bar, system menu icon, action bar, and scroll bar.

window coordinates In System/38 graphics, the user-defined set of coordinates mapped on the viewport from which the scale is drawn.

window edge The sequence number of the last data packet in a window (2).

window gravity In the AIX operating system, the attraction of a subwindow to some part of its parent. Window gravity causes subwindows to be automatically repositioned, relative to an edge, corner, or center of a window when resized. Synonymous with widget gravity.

Appellant: Cary L. Bates
Serial No.: 10/821,146
Filed: April 8, 2004
Title: Detecting Incorrect Versions of Files

Page 47

Docket: ROC920030419US1

10. RELATED PROCEEDINGS APPENDIX

None.